Piecewise Linear Models with Guaranteed Closeness to the Data

Longin Jan Latecki, Marc Sobel, and Rolf Lakaemper

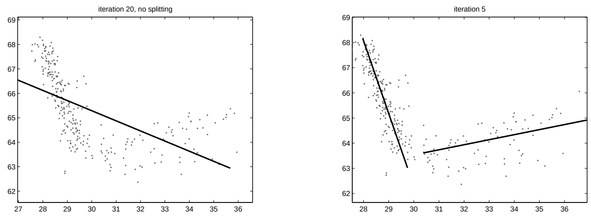
Abstract—This paper addresses the problem of piecewise linear approximation of point sets without any constraints on the order of data points or the number of model components (line segments). We point out two problems with the maximum likelihood estimate (MLE) that present serious drawbacks in practical applications. One is that the parametric models obtained using a classical MLE framework are not guaranteed to be close to data points. It is typically impossible, in this classical framework, to detect whether a parametric model fits the data well or not. The second problem is related to accurately choosing the optimal number of model components. We first fit a nonparametric density to the data points and use it to define a neighborhood of the data. Observations inside this neighborhood are deemed informative; those outside the neighborhood are deemed uninformative for our purpose. This provides us with a means to recognize when models fail to properly fit the data. We then obtain maximum likelihood estimates by optimizing the Kullback-Leibler Divergence (KLD) between the nonparametric data density restricted to this neighborhood and a mixture of parametric models. We prove that, under the assumption of a reasonably large sample size, the inferred model components are close to their ground truth model component counterparts. This holds independently of the initial number of assumed model components or their associated parameters. Moreover, in the proposed approach, we are able to estimate the number of significant model components without any additional computation.

Index Terms—Maximal Likelihood Estimate (MLE), Expectation Maximization (EM), Kullback-Leibler divergence (KLD), sparse EM, piecewise linear approximation.

1 Introduction

The main issue addressed by this paper is the relation between the number of model components and goodness of fit. Maximum likelihood estimators (MLEs) are one of the main statistical tools for assuring optimal parameter estimation. In this paper, we point out some serious drawbacks of parametric maximum likelihood estimators from the point of view of modeling data for practical applications. We propose an approach removing these drawbacks. It is illustrated by a line fitting example coming from the area of pattern recognition. Consider the set of points on the plane shown in Fig. 1(a). The best fitting line is computed using parametric maximum likelihood (which is equivalent to least squared fitting here). Note that the line does not fit the data properly. The approximation in Fig. 1(a) is bad, since the middle part of the line is not supported by the data (i.e., not close to any data points), and large portions of the data points are not in proximity to the line. It is obvious that the model in Fig. 1(b) is significantly better than that given in Fig. 1(a). Note that both segments in Fig. 1(b) are contained in the neighborhood of the data points.

The problem is related to the number of model components of the parametric density used in estimation. As we show below, it is frequently impossible to decide the optimal number of model components; this is a consequence of the presence of many local optima. Parametric maximum likelihood estimators correspond to the local optimum closest to the initial parameter values. Hence maximum likelihood estimates of parameters in mixture models depend on the initial values of the model parameters. Moreover, for a fixed number of model components, it is impossible to recognize bad models like the single line in Fig. 1(a) calculated using the (classical) MLE framework.



(a) (b)

Fig. 1. This figure illustrates the relation between the number of components and goodness of fit, which is the issue addressed by this paper. It is obvious to us that the approximation in (b) of the underlying data points is significantly better than the approximation in (a). (a) shows the best possible maximum likelihood approximation. (b) shows the result obtained by the proposed method.

Here we propose an approach that uses a nonparametric density estimate (NPDE) of the data points to define a neighborhood of the data points. Using this neighborhood we can easily recognize bad fits to the data. Any model component that is not contained in the aforementioned neighborhood is easily recognized

- *L. J. Latecki and R. Lakaemper are with the Dept. of Computer and Information Sciences at Temple University, Philadelphia.M. Sobel is with Statistic Dept. at Temple Univ.*
  E-mail: latecki@temple.edu

as providing a bad fit to the data, e.g., the estimated line in Fig. 1(a). Moreover, we are not only able to identify bad fits (between the model and the data) but also to improve on them. The main idea underlying this improvement is to keep only those parts of the model components that are contained in the aforementioned neighborhood. By doing so, we automatically provide an optimal assessment of the number of model components, and guarantee that the resulting model components are a good fit to the data.

Below, we discuss the second serious drawback of MLEs. The goal is to approximate the ground-truth density $q(x)$ with a member $p_\Theta(x)$ of a parametric family $\{p_\Theta(x) : \Theta \in \mathcal{S}\}$ of parametric mixtures of densities, where Θ is a vector of parameters from a parameter space $\mathcal{S}$. As is well-known, Kullback-Leibler divergence (KLD) can be used to measure dissimilarity between the ground-truth and parametric family of density mixtures. By definition, the KLD between the ground truth $q(x)$ and the density, $p_\Theta(x)$ is:

$$D(q(x)||p_\Theta(x)) = \int q(x)\log\frac{q(x)}{p_\Theta(x)}dx$$
$$= \int q(x)\log q(x)dx - \int q(x)\log p_\Theta(x)dx \quad (1)$$

It is easily shown that the parameters $\widehat{\Theta}$ minimizing (1) are given by

$$\widehat{\Theta} = \text{argmax}_\Theta\{\int q(x)\log p_\Theta(x)dx\} \quad (2)$$

The classical maximum likelihood estimator is obtained by applying the MC (Monte Carlo) integral estimator to (2) under the assumption that the observations $x_1, ..., x_n$ are i.i.d. (independently and identically distributed) sample points selected from the ground truth distribution $q(x)$.

$$\widehat{\Theta} = \text{argmax}_\Theta \sum_i \log p_\Theta(x_i) \quad (3)$$

We show below that the derivation of (3) from (2) is based on an assumption that is not satisfied in practical applications. The derivation of (3) follows from the fact that we can approximate the integral of a continuous function f (in our case, $f(x) = \log p_\Theta(x)$) by its Monte Carlo estimate if $x_1, \ldots, x_n$ are i.i.d. sample points drawn from the probability density function (pdf) $q(x)$.

$$\int f(x)q(x)dx \approx \frac{1}{n}\sum_i f(x_i) \quad (4)$$

In the usual approach to inference, it is a commonly accepted assumption that sample data points $x_1, \ldots, x_n$ are distributed according to the ground truth density $q(x)$. This assumption is the key to insuring that maximum likelihood estimators are appropriate for purposes of estimating parameters of interest. However, this assumption is incorrect in practical applications, since real datasets are usually noisy. This noise cannot be properly characterized by the ground truth density $q(x)$. As a consequence, we cannot properly model the data using the ground truth density $q(x)$.

We propose a solution to this problem in Section 2. We demonstrate that the ground-truth density $q(x)$ can be estimated from the data using a nonparametric density estimate. This allows us to use Kullback-Leibler divergence (KLD) to fit an optimal model to this estimate rather than to the noisy data.

There is a close relation between the unrealistic derivation of (3) from (2), and the estimation of the number of model components. When using KLD it is possible to estimate the optimal number of significant model components of p_Θ. This is due to the fact that KLD $D(q||p_\Theta)$, viewed as a functional on the space $\{p_\Theta\}$ of Gaussian mixtures, is convex and hence has a minimum; this minimum does not have to be a finite mixture of Gaussians, since the space of finite Gaussian mixtures is not closed. The set of finite Gaussian mixtures is dense in the space of continuous functions. Therefore, we can estimate the minimum with any required precision when we minimize KLD in the space of finite Gaussian mixtures. In particular, this means that we can estimate the number of significant mixture components, although it is impossible to determine precisely the optimal number of components, since this number may be large or infinite (e.g., some ground truth model components could be very small). Therefore, we use KLD to estimate the number of *significant* model components.

When optimizing the log likelihood in formula (3), we cannot estimate the number of model components. It is a known fact that the log likelihood function (3) increases when the number of model components is increased. The unrealistic derivation of (3) from (2), explains why the ability to estimate the number of significant model components using KLD is lost when we adopt the MLE framework (3).

We derive a properly weighted version of maximum likelihood estimation from (2) in Section 2 and demonstrate experimentally that it accurately estimates the number of significant model components. We also show that in the proposed framework, our version of EM converges to an optimal solution even if the initial values of model parameters are not close to being globally optimal.

As we prove in Section 4, our model estimates are close to the ground truth. This is a consequence of the fact that we minimize the Kullback-Leibler divergence between the ground truth and those models. This result does not depend on the number of model components or parameter values assumed at the outset. However, the question arises regarding how to actually minimize the KLD; in particular, how to properly adjust the number of mixture components. The sparse EM algorithm proposed in Neal and Hinton [9] provides a nice framework for adjusting the number of model components. The algorithm of Neal and Hinton allows us to freeze the probabilities for most values of hidden variables, and recompute the probabilities for only a small fraction of ’plausible’ values. The values of hidden variables range over the indices of model components. We observe here that this framework also allows us to change the number of ’plausible’ values, which is tantamount to changing the number of model components assumed. For example, we can (i) freeze the probabilities for all values of hidden variables except those of two model components, and then (ii) replace the two indices with a single new index if KLD is smaller. This effectively merges two model components into a single component.

Our approach is presented in a unified mathematical framework with a single target function (KLD) that is optimized in classical E and M steps as well as in the proposed split, merge, and component insertion steps, which are needed to adjust the number of model components. The results given in [9] guarantee the convergence of our algorithm. In comparison to the classical EM algorithm, based on the log likelihood of the data, the proposed algorithm is less susceptible to converging to locally optimal estimators which are not also globally optimal. Merge, split, and insert steps, when taken, explicitly improve the quality of the inferred model. As a consequence, in contrast with the classical EM algorithm, we are able to take care of egregious outliers. We stress that it is not possible to accurately estimate the number of model components in the original approach of Neal and Hinton [9].

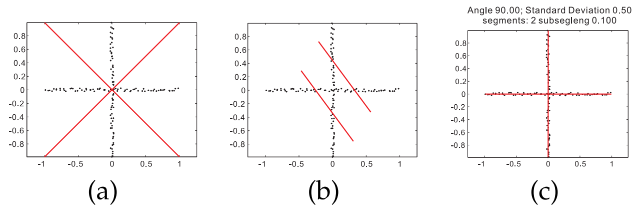
There exist many possible applications that require optimal adjustment of model components. We focus on polygonal approximation in this paper. We note that the proposed framework has a broader scope of possible applications. We illustrate our approach on polygonal approximation of point sets forming curves in digital images. In [7] we demonstrate the application of the proposed approach to fitting line segments to laser range data. An overview of techniques for polygonal approximations of curves, which require that the order of data points is known, can be found in [10].

The main difficulty of fitting polylines in such applications is that the segmentation (or correspondence) of data points to line segments as well as the order of data points is unknown. The Expectation Maximization (EM) algorithm [3] provides a particularly useful framework to solve this correspondence problem. The use of the EM algorithm for purposes of line fitting is known as the Healy-Westmacott procedure in statistics, and predates the EM algorithm by many years [5]. However, polygonal approximation of point data requires preliminary estimates of the model parameters and the number of model components (line segments) but, as observed above, in the classical EM framework the number of model components must be known and fixed in advance.

The failure of maximum likelihood estimates in parametric mixture models to provide a good fit to the data has not yet received the attention it deserves in the literature. In contrast, the problem of characterizing the optimal number of model components has received a significant amount of attention in the literature, see e.g., [1]. A correct characterization of the number of components and their parameter values for a statistical model is crucial in all EM applications. This task is made more difficult as a consequence of the propensity of algorithms to get stuck in local optima; as a result, this problem is one of the most challenging in statistical reasoning.

EM extensions proposed in the literature that estimate the number of model components are based on split and merge steps applied to the existing model components. We observe that the existing split and merge approaches (e.g., Green [4] and Ueda et al. [14]) cannot be guaranteed to correctly estimate the optimal number of model components due to the fact that they cannot distinguish between locally optimal and globally optimal solutions. The approach presented in [13] explicitly estimates the distribution of model parameters and the number of model components in an extended EM framework. However, this approach as well as the approaches in [4], [14] do not guarantee that the model components are close to the data points.

Even if the number of model components is properly estimated, EM may not yield a globally optimal solution. We give a simple example that illustrates the fact that EM yields a locally but not globally optimal solution if the initial values of the model parameters are not close to their globally optimal values in Fig. 2. We observe again that they can be recognized by the fact that parts of the lines are not close to the data points.



(a) (b) (c)

Fig. 2. The data points follow horizontal and vertical lines in a cross-like pattern. (a) and (b) show two locally optimal approximations of the data points obtained by the classical EM algorithm for two different initial positions of two lines. (c) shows the optimal approximation obtained using the proposed method on the same input.

2 Optimizing KLD

In all applications, the sample data points are corrupted by a certain amount of noise. Usually the proportion of noisy points does not decrease when the number of sample points is increased. We quantify this corruption by assuming that the data follow a distribution consisting of a mixture of an unknown ground-truth distribution $q(x)$ and an unknown noise distribution $\eta(x)$. Let $u(x) = \alpha q(x) + (1-\alpha)\eta(x)$ denote this mixture distribution. The quantity, α is the probability that an observation comes from the ground-truth distribution $q(x)$ and $(1-\alpha)$ is the probability that it comes from the noise distribution. Since the observed sample data points follow the noisy distribution $u(x)$ rather than the ground truth distribution $q(x)$, we obtain a more accurate Monte Carlo estimate of the integral in (4):

$$\int f(x)q(x)dx = \frac{\int f(x)q(x)dx}{\int q(x)dx} = \frac{\int f(x)\frac{q(x)}{u(x)}u(x)dx}{\int \frac{q(x)}{u(x)}u(x)dx} \approx \frac{\sum_i f(x_i)\frac{q(x_i)}{u(x_i)}}{\sum_i \frac{q(x_i)}{u(x_i)}} \tag{5}$$

The ratio

$$\frac{\alpha q(x)}{u(x)} = \frac{\alpha q(x)}{\alpha q(x) + (1-\alpha)\eta(x)} \tag{6}$$

can be interpreted as the conditional probability, $P(ground\ truth|x)$, that an observed data point x is selected from the ground truth density $q(x)$. We note that large values of $P(ground\ truth|x)$ indicate that the data point x is of significant interest for inferential purposes; small values indicate the reverse.

We define a **smoothed data density** $sdd(x)$ as

$$sdd(x_i) = \frac{\frac{q(x_i)}{u(x_i)}}{\sum_i \frac{q(x_i)}{u(x_i)}} \tag{7}$$

By plugging sdd into (5), we obtain one of our key equations

$$\int f(x)q(x)dx \approx \sum_i f(x_i)sdd(x_i) \tag{8}$$

In Section 3 we show that the estimate of sdd is equal to the estimate of the ground-truth density q. In the proposed framework, (8) replaces (4). It can be easily shown that (8) leads to a substantially smaller mean squared error in the estimation of the integral than (4). Consequently, if some proportion of the observations $x_1, ..., x_n$ are noisy, we obtain from (8) that a more accurate estimator of Θ in (2) is given by:

$$\widehat{\Theta} = \text{argmax}_\theta \sum_i \log p_\theta(x_i)sdd(x_i). \tag{9}$$

In the proposed framework, (9) replaces (3). It is well known that (3) cannot be used to estimate the correct number of model components, since (3) increases when the number of model components increases. In contrast, we are able to determine the correct number of significant model components by (9). Thus, the modified EM algorithm that maximizes (9) accurately estimates both the right number of significant model components and their associated model parameters.

In Fig. 3 we illustrate the difference between the proposed (9) and the classical equation (3) in the case of classical maximum likelihood estimators when the parametric model specifies a single line segment. Fig. 3 shows a real data set obtained from a microscopic analysis of a wafer. The visible curve is a scratch that needs to be detected. All other points are background noise. The initial line segment is shown in red in (a). The result of the classical MLE estimate based on (3) is shown in (b). The proposed MLE estimate based on (9) is shown in (c). The superior result of (9) can be intuitively explained by the fact that sdd downweights noisy points. Therefore, the line segment is able to reach the globally optimal position in (c). Observe that it is impossible to obtain the result in (c) with any distance-weighted regression [11], with point weights computed based on distances to the model line. In contrast, the weights of points in our approach are based on a nonparametric density estimate and thus more accurately reflect the spatial density of the data points (see Section 3).

In order to better fit the scratch curve, we need more line segments, which requires that we identify the optimal number of model components. Our approach for achieving this goal is described in the rest of this paper. The experimental results are presented in Section 7.

3 Estimating the Data Density and the Informative Data Points

In this section we show that sdd is equal to the estimate of the ground-truth density q, i.e., $sdd \approx \widehat{q}$. We recall that $u(x)$ is the density of the observed data. Following the assumptions made in calculating bootstrap samples, we can take the value of the density, $u(x)$ at the observed i.i.d. sample points $x_1, \ldots, x_n$ drawn from $u(x)$ to be $\widehat{u}(x_1) = \cdots = \widehat{u}(x_n) = \frac{1}{n}$. Thus, estimating the ratio (6) reduces to estimating the ground truth density $q(x)$:

$$\frac{q(x_j)}{u(x_j)} \approx n\widehat{q}(x_j). \tag{10}$$

Since $sdd(x_j) \propto \frac{q(x_j)}{u(x_j)}$, and both sdd and $\widehat{q}$ are normalized (to sum to 1) we obtain $sdd \approx \widehat{q}$.

We use kernels to estimate the ground truth distribution; these are the most widely used nonparametric density estimation method. There is a large body of published literature on nonparametric density estimation. One of the most efficient approaches is the class of variable width kernel density estimators defined by,

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{h(x_i)^d} K(\frac{x - x_i}{h(x_i)}) \tag{11}$$

where d is the dimension of the Euclidean space, K is a kernel function, which we take to be Gaussian, and $h(x_1), \ldots, h(x_n)$ are the bandwidths defined at the respective data points $x_1, , x_n$. One of the main advantages of (11) is that if K is a density, then so is f [12]. The simplest version of the function h is a constant function $h(x_i) = h$, where h is a fixed bandwidth. However, for the kind of datasets met in practice, local sample densities may vary, e.g., a robot scans one part of the wall more frequently than the other parts. Therefore, it is necessary to have a method that is adaptive to changes in local density. This adaptation is obtained, for example, by taking $h(x_i) = hd_k(x_i)$, where d_k is the distance to kth nearest neighbor of point x_i. The use of kth nearest

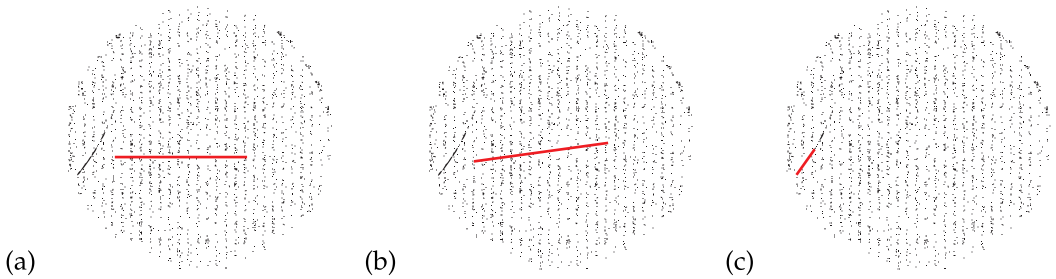


Fig. 3. A real data set from a microscopic analysis of a wafer. The visible curve is a scratch that needs to be detected. All other points are background noise. The initial line segment is shown in red in (a). The result of the classical MLE estimate based on formula (3) is shown in (b). The proposed MLE estimate based on formula (9) is shown in (c).

neighbors for this purpose was first proposed in [8], see also [2]. In our approach, we compute (11) in two steps. First we estimate, the weight at each sample point x_j

$$w(x_j) \propto f(x_j) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{h(x_i)^d} K(\frac{x_j - x_i}{h(x_i)}), \qquad (12)$$

where proportionality refers to the fact that $\sum w(x_i) = 1$. We do not attempt to estimate $\widehat{q}$ using (12), but instead we estimate $\widehat{q}$ by convolution of f with a Gaussian kernel ϕ

$$\widehat{q}(x) = \frac{1}{h^d}\sum_{i=1}^{n} \phi(\frac{x - x_i}{h}) f(x_i) = \sum_{i=1}^{n} \frac{w(x_i)}{h^d} \phi(\frac{x - x_i}{h}). \qquad (13)$$

Equation (13) amounts to using a larger bandwidth of $\sqrt{h^2 + h(x_i)^2} = h\sqrt{1 + d_k(x_i)^2}$ in (11). The convolution in (13) makes sense intuitively, since the observed density is not the ground-truth density q but a noise corrupted density $u(x) = \alpha q(x) + (1-\alpha)\eta(x)$. Therefore, the convolution with a Gaussian kernel allows us to downweight the noise component η, and consequently, more accurately estimate the ground-truth density q.

We define the neighborhood $U(X)$ of the data points $X = x_1, \ldots, x_n$. This is a central concept of the proposed approach. The intuition behind it is that, instead of calculating maximum likelihood estimates based on the entire observation space D, we restrict attention to a subset $U(X) \subset D$ that includes sufficiently informative regions of the data points X. Thus, from the topological point of view $U(X)$ is the neighborhood of only a subset of the full set X of data points. We assume that the observation space is $D = \Re^d$, where $\Re$ are the real numbers, and $d = 1, 2, \ldots$.

Since the values of $\widehat{q}$ are extremely small outside a bounded region around the dense part of the sample points $X = x_1, \ldots, x_n$, we define the neighborhood $U_\delta(X) = \{x \in D : \widehat{q}(x) \geq \delta\}$. We note that $U_\delta(X)$ is a compact set, since it is closed and bounded.

As is demonstrated below, maximum likelihood estimators restricted to $U_\delta(X)$ perform significantly better than their unrestricted counterparts. Restricting EM to $U_\delta(X)$ allows us to prevent convergence to local optima which are not global optima. This is a consequence of the fact that the uninformative parts of model components (lying outside the compact neighborhood $U(X)$) are removed. For example, convergence to the locally optimal solution in Fig. 1(a) is impossible in our approach, since the middle part of the line segment is removed; effectively splinting one model component to two. Consequently, only parts of the two line segments around the sample points remain. This allows to correctly reposition the two remaining line segments. We obtain the model with two components (line segments) with good fit to the data as shown in Fig. 1(b).

4 Global Convergence

We prove that the parametric models generated by the proposed approach converge in the observation space to the ground-truth model. Convergence in the observation space means that the estimated model components are close to their ground-truth component counterparts. As is illustrated in Fig. 2, this result does not hold for the classical EM algorithm even if the number of model components is correct. We note that the convergence in the observation space does not imply convergence in the parameter space, i.e., the parameters of the estimated model components may not be close to the parameters of the ground truth components. Usually statisticians talk about convergence in the parameter space; we focus here on convergence in the observation space. We stress that this aforementioned convergence leads to improved convergence in the parameter space. We demonstrate this fact with experimental results.

We impose a hard constraint on the estimated parametric model; we suppose that each of its model components is contained in the set $U_\delta(X)$ (defined in Section 3). The hard constraint restricts the domain over which we estimate the parametric density.

Let $s_1, \ldots, s_k$ be line segments that generate the ground truth model components, and let q be a ground truth distribution generated from this model. We assume that the distribution of sample points along line segments is sufficiently dense (i.e., there are no sample point gaps along the ground truth line segments). We assume that data points have a Gaussian likelihood as function of their distances to the model components with a common standard deviation σ. (The assumption of a common σ is not essential, i.e., we can have a different standard deviations for each model component.)

We define a topological ϵ neighborhood of the model components to be $U_\epsilon(\bigcup_{j=1}^k s_j)$; this is the set of all point whose minimum distance to the model components is less or equal to ϵ.

Theorem. For any $\epsilon > 0$, there exists $\delta > 0$ such that, for sufficiently large number of sample points n, with probability approaching 1,

$$\bigcup_{j=1}^{\widehat{k}} \widehat{s}_j \subseteq U_\epsilon(\bigcup_{j=1}^{k} s_j)$$

for every parametric model p_Θ with model components determined by the line segments $\widehat{s}_1, \ldots, \widehat{s}_{\widehat{k}}$ estimated by our approach from sample points $x_1, \ldots, x_n$.

Proof: Let $\epsilon > 0$. By choosing δ so that, $\delta = \phi_\sigma(\epsilon)$ where ϕ_σ is a Gaussian with mean zero and std σ, we have that $q(x) \geq \delta$ implies $x \in U_\epsilon(\bigcup_{j=1}^k s_j)$ for any point x.

The key observation is the fact that the nonparametric density estimate $\widehat{q}$ approaches the ground truth distribution q as the sample size n increases (this refers to convergence in the function space of pdfs). Therefore, if $x \in U_\delta(X)$, i.e., $\widehat{q}(x) \geq \delta$ we obtain, with probability approaching one, that $q(x) \geq \delta$. This implies that $x \in U_\epsilon(\bigcup_{j=1}^k s_j)$. Thus, we have shown that, with probability approaching one, $U_\delta(X) \subseteq U_\epsilon(\bigcup_{j=1}^k s_j)$.

Let p_Θ be the parametric model generated by our approach with model components determined by line segments $\widehat{s}_1, \ldots, \widehat{s}_{\widehat{k}}$. Since we impose the hard constraint that each line segment representing a model component is contained in the set $U_\delta(X)$, any part of any line segment outside $U_\delta(X)$ is removed (see the split step defined in Section 6). Consequently, we have $\bigcup_{j=1}^{\widehat{k}} \widehat{s}_j \subseteq U_\delta(X)$. It follows that with probability approaching one $\bigcup_{j=1}^{\widehat{k}} \widehat{s}_j \subseteq U_\epsilon(\bigcup_{j=1}^k s_j)$ as the sample size n goes to infinity. This proves the theorem.

As illustrated in Figs. 1(a) and 2(a,b), this theorem does not hold for line models fitted in the classical EM framework.

5 E and M Steps

Beginning with this section we present an extension of the standard EM framework that allows us to compute the proposed MLE estimate in formula (9). We also show that we optimize the same target function in E and M steps as in component split and merge steps.

We introduce latent variables $z_1, ..., z_n$ which serve to properly label the components of the respective data points $x_1, ..., x_n$. It is assumed that the pairs (x_i, z_i) for $i = 1, \ldots, n$ are i.i.d. with common (unknown) joint (ground truth) density, $q(x,z) = q(x)q(z|x)$; $q(x)$ is the marginal x-density and $q(z|x)$ is the conditional density of the label z given x. In this new framework, the KLD between the joint density $q(x,z)$ and a parametric counterpart density $p_\Theta(x,z)$ is

$$\begin{aligned} &D(q(x,z)||p_\Theta(x,z)) = D(q(x)q(z|x)||p_\Theta(x)p_\Theta(z|x)) \\ &= \int_x \int_z \left\{ \log\left[\frac{q(x)}{p_\Theta(x)}\right] + \log\left[\frac{q(z|x)}{p_\Theta(z|x)}\right] \right\} q(x)q(z|x)dzdx \\ &= \int_x \log\left[\frac{q(x)}{p_\Theta(x)}\right] q(x)dx + \int_x q(x) \int_z \log\left[\frac{q(z|x)}{p_\Theta(z|x)}\right] q(z|x)dz \end{aligned} \tag{14}$$

We are now ready to introduce the expectation (E) and maximization (M) steps. Both steps aim at minimizing the same target function (14) in our framework. The expectation step yields the standard EM formula; considerations discussed above lead to a different solution for the maximization step.

Expectation Step: For a fixed set of parameters Θ, we want to find a conditional density $q(z|x)$ that minimizes $D(q(x,z)||p_\Theta(x,z))$. Since KLD is always nonnegative, and the second summand in (14) is minimized for $q(z|x) = p_\Theta(z|x)$ (in which case it is equal to zero), we obtain from (14) that

$$q(z|x) = p_\Theta(z|x) \text{ minimizes } D(q(x,z)||p_\Theta(x,z)).$$

In particular, for given sample points $x_1, \ldots, x_n$, we obtain

$$\begin{aligned} &q(z_i = l|x_i) = p_\Theta(z_i = l|x_i) = p(z_i = l|x_i, \Theta) \\ &= \frac{p(x_i|z_i = l, \Theta)p(z_i = l|\Theta)}{p(x_i|\Theta)} = \frac{p(x_i|z_i = l, \Theta)\pi_l}{\sum_{j=1}^k p(x_i|z_i = j, \Theta)\pi_j}, \end{aligned} \tag{15}$$

where $\pi_l = p(z_i = l|\Theta)$ and $\pi_j = p(z_i = j|\Theta)$ are the prior probabilities of component labels l and j respectively.

Maximization Step: For the fixed marginal distribution $q(z|x) = p_\Theta(z|x)$, we want to find a set of parameters Θ that minimizes (14). Substituting $q(z|x) = p_\Theta(z|x)$ in (14), we obtain

$$D(q(x,z)||p_\Theta(x,z)) = \int \log(\frac{q(x)}{p_\Theta(x)})q(x)dx = D(q(x)||p_\Theta(x)) \tag{16}$$

Thus, minimizing $D(q(x,z)||p_\Theta(x,z))$ in Θ is equivalent to minimizing $D(q(x)||p_\Theta(x))$ in Θ. Using the estimate derived in equation (9), minimizing (16) in Θ is equivalent (in the MC setting discussed above) to maximizing the weighted marginal density

$$\begin{aligned} WM(\Theta) &= \sum sdd(x_i) \log p_\Theta(x_i) \\ &= \sum_{i=1}^n sdd(x_i) \log[\sum_{l=1}^k p(x_i|z_i = l, \Theta)p(z_i = l|\Theta)] \\ &= \sum_{i=1}^n sdd(x_i) \log[\sum_{l=1}^k p(x_i|z_i = l, \Theta)\pi_l] \end{aligned} \tag{17}$$

where $\pi_l = p(z_i = l|\Theta)$ are the prior probabilities of component labels $l = 1, \ldots, k$.

We explicitly use the incremental update steps of the EM framework. Using the prior probabilities of component labels $\pi_l^{(t)} = p(z_i = l|\Theta^{(t)})$ obtained at stage t for

$l = 1, ..., k$, we obtain from (17) that an update of $WM(\Theta)$ is estimated by maximizing

$$WM(\Theta;\Theta^{(\mathrm{t})}) = \sum_{i=1}^{n} sdd(x_i)\log[\sum_{l=1}^{k} p(x_i|z_i = l,\Theta)\pi_l^{(t)}] \tag{18}$$

in Θ with $\Theta^{(t)}$ denoting the value of Θ computed at stage t of the algorithm. The crucial difference between this and the standard EM update is that our target function is weighted with terms $sdd(x_i)$. We note that the known convergence proofs for the EM algorithm apply in our framework, since appending the weights $sdd(x_i)$ in (18) does not influence the convergence.

6 Split, Merge, and Comp. Insertion

We introduce new split, merge, and component insertion steps needed to adjust the number of model components. We stress that we always optimize the same target function (18) when performing these steps. This means that each of these steps is performed only if the value of the target function (18) is improved.

Split: We remove parts of the model components that lie outside $U_\delta(X)$. Thus, for a given model component (line segment) l, the split is performed by calculating the intersection $l \cap U_\delta(X)$. This may yield, (i) new model components which consist of shorter subsegments of l, or (ii) the segment l may be removed entirely. For simplicity of presentation, assume that component l is split into two disjoint components, i.e., $s_{l_1} \cup s_{l_2} = s_l \cap U_\delta(X)$, where s_j is a geometric entity representing model component j (e.g., s_j is a line segment in our applications). Then we compute the new parameters for the components l_1, l_2 by performing sparse E and M steps [9].

In the sparse E step, we freeze all posterior probabilities $p(z_i = j|x_i, \Theta)$ for $j \neq l$, where $j \in \{1, \ldots, k\}$ is the index of the model components existing at step t, and compute the posterior probabilities $p(z_i = j|x_i, \Theta)$ for $j = l_1, l_2$.

In the sparse M step, we only recompute the parameters of components l_1, l_2 based on the probabilities computed in the sparse E step. We obtain the parameters maximizing the equation (18) by differentiating (18) with respect to these parameters. The new parameters obtained are for lines that we need to trim to get the line segments. The trimming results from selecting line segments which belong to the intersection of the lines with the set $U_\delta(X)$. This is computed by placing sample points on a line (with a sufficient density), and computing the values of sdd at the sample points. Then parts of the line with sample points having sdd values below δ are removed. Then component l is replaced of with components l_1, l_2. This replacement does not decrease the value of our target function (18), since in the worst case components l_1, l_2 will be aligned with parts of component $l \subset U_\delta(X)$.

Merge: Given a candidate component l, we merge two existing model components l_1, l_2 to l if for $j \in \{1, \ldots, k\}$

$$\begin{aligned} WM(\Theta;\Theta^{(t)}) &= \sum_{i=1}^{n} sdd(x_i)\log[\sum_{j} p(x_i|z_i = j,\Theta)\pi_j^{(t)}] \\ &> \sum_{i=1}^{n} sdd(x_i)\log[\sum_{j\neq l} p(x_i|z_i = l,\Theta)\pi_l^{(t)} \\ &+ p(x_i|z_i = l_1,\Theta)\pi_{l_1}^{(t)} + p(x_i|z_i = l_2,\Theta)\pi_{l_2}^{(t)}] \end{aligned} \tag{19}$$

We only need to perform sparse computations to perform the merge test. We need to compute the corresponding probabilities for the candidate component l, subject to the constraint $\pi_l^{(t)} = \pi_{l_1}^{(t)} + \pi_{l_2}^{(t)}$. If (19) holds and we replace l_1, l_2 with l, the convergence of our algorithm follows from the results of Neal and Hinton [9]. The parameters are estimated after the sparse EM step in equation (15) in [9] is taken. Since such local computations are performed only if the target function increases, our algorithm is guaranteed to converge. (Here we talk about local convergence in the parameter space.)

Component Insertion: Assume that the model is composed of k components, and we consider adding a component $k+1$. Since our goal is maximizing $WM(\Theta;\Theta^{(t)})$ in formula (18), we simply need to check whether adding a component $k+1$ increases WM, where $j \in \{1, \ldots, k\}$:

$$\begin{aligned} WM(\Theta;\Theta^{(t)}) &= \sum_{i=1}^{n} sdd(x_i)\log[\sum_{j=1}^{k} p(x_i|z_i = j,\Theta)\pi_j^{(t)}] \\ &< \sum_{i=1}^{n} sdd(x_i)\log[\sum_{j=1}^{k+1} p(x_i|z_i = j,\Theta)\pi_j^{(t)}] \end{aligned} \tag{20}$$

Prior to evaluating this inequality but after adding the new component, we perform the E step in order to compute the corresponding probabilities for the components $1, \ldots, k, k+1$ after the new component $k+1$ has been added. Thus, the probabilities and the components weights $\pi_1^{(t)}, \ldots, \pi_k^{(t)}$ on the right hand side of (20) may have different values than those on the left hand side.

Our framework is very general in that it allows many possible selections of the candidate components for the component insertion step. We use a simple heuristic to obtain new components for insertion. We assign each data point to the most likely component. A new component is computed with a weighted regression that fits a single line to all data points that are not sufficiently close to the assigned components.

The component insertion step yields a natural stop criterion of the proposed algorithm. The algorithm terminates if all inserted components are removed by split and merge steps. A component removal by a split step means that a new component does not have sufficient support in the data points, while the removal by a merge step means that it is very similar to one of the existing components.

7 Applications

Since we added split, merge, and component insertion steps in addition to the E and M steps, the question arises regarding what is the best order of these steps. To produce the experimental results presented in this section, we used the following order, which was determined experimentally: component insertion, split, E, M, and merge. We call the sequence of these five steps one iteration of our algorithm. Clearly, we first computed sdd for a given data set before these steps were applied.

Our first application involves finding scratches and other defects in data obtained from microscopic analysis of wafers. In this application there is no initial estimate of a possible position of defects, and the number of background noise points is significantly larger than the number of signal points. Additionally, the point density of the signal as well as that of the noise varies significantly for different images. Fig. 4 shows some example input images (left) and our results as line segments superimposed on the input images (right). In all experiments we initialized our algorithm with two crossed line segments like the ones shown in top left of Fig. 4. The red line segments (right) show the final results obtained by our algorithm after only five iterations. We obtained results of this quality on over 100 test images of this kind. Some of the test data sets did not contain any signal points, and in this case, our algorithm correctly did not fit any line segments. The number of data points for each wafer vary from few hundred to several thousand points. The numbers of input data points and the output line segments for our examples are shown in the caption. We used the same parameters for all test images: $h = 5$, $\sigma = 0.5$, and $\delta = 10^{-7}$ for sets with more than 2,000 points and $\delta = 10^{-6}$ for sets with the smaller number of points. Our Matlab implementation needed between one and two minutes for each image on Pentium 4 CPU 3.2GHz. About half of the total CPU time was spent on KDTree indexing.

Our second application is learning one-dimensional manifolds. We sampled 10,000 points from a normal distribution around a spiral in Fig. 5. Fig. 5(a) shows two initial model components. Fig. 5(b) shows an approximation obtained after two iterations. Fig. 5(c) shows the final approximation obtained after only five iterations. Our algorithm automatically determined the shown 50 line segments as model components. By comparison, the SMEM algorithm in [14] needed 353 iterations to converge to a simpler spiral. We used parameters: $h = 0.5$, $\sigma = 0.05$, and $\delta = 0.00005$.

Figs. 5(a,b) illustrate the advantage of the proposed approach. In this setting we have parts of model components (i.e., line segments) that lie in the regions of the observation space containing no data points. Our algorithm simply removes these parts. This limits the observation space to the set $U_\delta(X)$, which has the effect of splitting the model components. In contrast, the SMEM algorithm ([14]) and the classical EM algorithm may generate model components that are not supported by data points, which then are trapped in local optima.

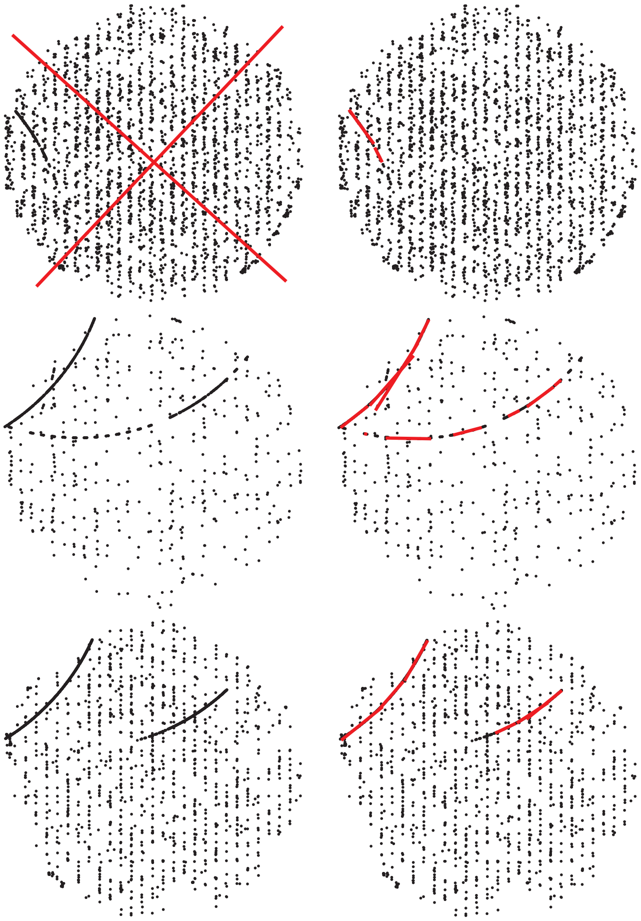


Fig. 4. We applied our algorithm to finding scratches and other defects in data obtained from microscopic analysis of wafers. (left): input image, (right): the obtained line segments in red overlaid over the input image. The numbers of input points and output line segments are (row 1): 4128 → 6, (row 2): 1543 → 10 (row 3): 5463 → 7.

8 Conclusions

We propose to infer parametric models from nonparametric density estimates constructed from the data using kernel density functions. The proposed approach has several advantages in comparison to classical EM approaches that infer parametric models directly from the data. We proved that the model components obtained by our algorithm are close to the ground-truth model components in the observation space. This implies that the model components obtained by our algorithm are guaranteed to be close to the data points. This fact entails many strong convergence properties of model components in the parameter space, and consequently, improves goodness of fit. It is also possible to infer the number of significant model components in the proposed approach. Moreover, the convergence of the inferred parametric models do not depend on the initial

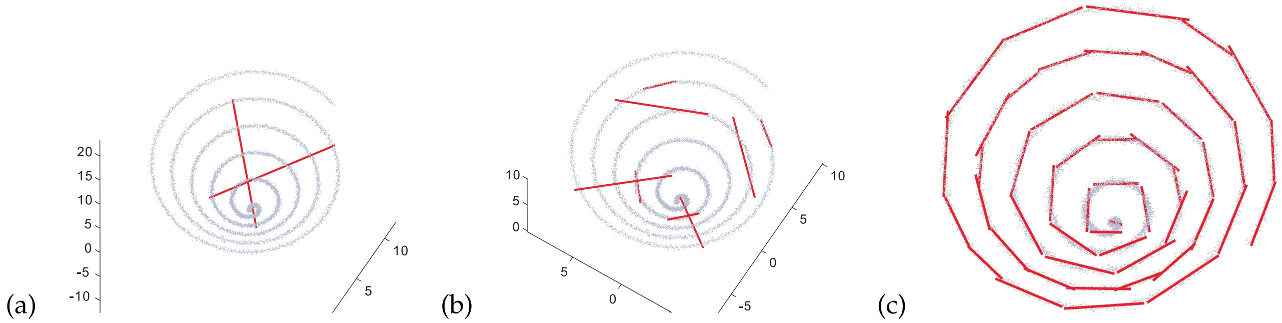


Fig. 5. (a) 10,000 points sampled from a normal distribution around a spiral in 3D, and two line segments as initial model components. (b) An approximation obtained after two iterations. (c) The final approximation obtained after only five iterations with 50 line segments shown in red.

values and the number of the model parameters. We demonstrate these properties with experimental results on several real and simulated data sets, in which the model components correspond to line segments. Our approach is presented in a unified mathematical framework with a single target function that is optimized in classical E and M steps as well as in the proposed split, merge, and component insertions steps, which are needed to adjust the number of model components. We use KLD to compare the parametric and nonparametric densities. The main challenge faced by our approach is the estimation of the nonparametric density from the data points.

We have focused on using the proposed algorithm to fit line segments. However, it can be extended to fit planar polygons to point clouds in 3D. Examples illustrating fitting planar polygons to 3D range data obtained from a moving robot are given in [6].

ACKNOWLEDGMENT

This work was supported in part by the NSF Grants IIS-0534929, IIS-0812118, and by the DOE Grant DE-FG52-06NA27508. We would like to thank Guy Shechter for providing his KDTREE software in Matlab and for extending it by a KDNN function, which was extremely useful for our experimental results.

REFERENCES

[1] M. J. Beal and Z. Ghahramani. The variational bayesian em algorithm for incomplete data. In J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. F. M. Smith, and M. West, editors, *BAYESIAN STATISTICS 7*. Oxford Univ. Press, 2003.

[2] L. Breiman, W. Meisel, and E. Purcell. Variable kernel estimates of multivariate densities. *Technometrics*, 19(2):135–144, 1977.

[3] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.

[4] P. Green. Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrica*, 82:711–732, 1995.

[5] M. J. R. Healy and M. Wesmacott. Missing values in experiments analyzed on automatic computers. *Appl. Statist.*, 5:203–206, 1956.

[6] R. Lakaemper and L. J. Latecki. Decomposition of 3d laser range data using planar patches. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2006.

[7] L. J. Latecki, M. Sobel, and R. Lakaemper. New EM derived from Kullback-Leibler divergence. In *ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD)*, 2006.

[8] D. O. Loftsgaarden and C. P. Quesenberry. A nonparametric estimate of a multivariate density function. *Ann. Math. Statist.*, 36:1049–1051, 1965.

[9] R. Neal and G. Hinton. A view of the em algorithm that justifies incremental, sparse, and other variants. In M. I. Jordan, editor, *Learning in Graphical Models*. Kluwer, 1998.

[10] P. L. Rosin. Techniques for assessing polygonal approximations of curves. *IEEE Trans. PAMI*, 19(3):659–666, 1997.

[11] T.P. Ryan. *Modern Regression Methods*. Wiley, New York, 1997.

[12] G. R. Terrell and D. W. Scott. Variable kernel density estimation. *The Annals of Statistics*, 20(3):1236–1265, 1992.

[13] N. Ueda and Z. Ghahramani. Bayesian model search for mixture models based on optimizing variational bounds. *Neural Networks*, 15:1223–1241, 2002.

[14] N. Ueda, R. Nakano, Z. Ghahramani, and G. E. Hinton. Smem algorithm for mixture models. *Neural Computation*, 12(9):2109–2128, 2000.